A Comparative Study on Ant-Colony Algorithm and Genetic Algorithm for Mobile Robot Planning

Piraviendran a/l Rajendran and Muhaini Othman(✉)

Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat Johor, Malaysia
muhaini@uthm.edu.my

Abstract. This paper investigates the optimization of routing for robotics vehicles in automated warehouses in Malaysia. Focusing on routing optimization, the study evaluates Ant-Colony Optimization (ACO) and Genetic Algorithm (GA) in Mobile Robot Planning. Key challenges includes efficient routing among task scheduling, and path planning complexities. Objectives include analyzing features of mobile robot planning and representing them in ACO and GA, implementation ACO and GA algorithms for solving routing problems using dataset, and evaluating their performance. The research anticipates significant contributions to algorithmic solutions, utilizing Python-based experiments aligned with Software Engineering practice, providing practical insights for routing optimization in automated warehouses. Results indicates that ACO outperforms GA in minimizing travel distance, establishing it as the superior routing algorithm for both case studies. Case study 1, the ACO algorithm achieved a best distance of 1036 (u) with execution time 1.67 (s), while the GA algorithm resulted in a best distance 1062 (u) with execution time 0.08 (s). For case study 2, the ACO algorithm achieved a best distance of 1071 (u) with execution time 1.91 (s), while the GA algorithm resulted in a best distance of 1082 (u) with execution time 0.08 (s). Multiple code execution cycles are conducted to provide average findings, ensuring the strength and consistency of the assessment. In conclusion, the study successfully identifies key features in warehouses routing, implements ACO and GA algorithms, and evaluates the performance based on achieved routes and distance.

Keywords: Automated Warehouse · Ant-Colony Optimization (ACO) · Genetic Algorithm (GA) · Routing Optimization · Mobile Robot Planning · Travel Distance · Execution Time

1 Introduction

Over the past five decades, robotics has undergone significant evolution, particularly with the widespread adoption of industrial robots since the 1960s [1]. Robotics has diversified into various classifications, playing a pivotal role in Industry 4.0 across multiple sectors [2–4]. Warehousing has transitioned from manual handling to automation, facilitated by smart automated warehouses integrating IoT, AI, and data analytics [5–8].

R. Ghazali et al. (Eds.): SCDM 2024, LNNS 1078, pp. 286–295, 2024.
https://doi.org/10.1007/978-3-031-66965-1_28

Autonomous guided vehicles, crucial in automating material handling processes within warehouses, have seen significant adoption, particularly in Malaysia with the rise of Automated Storage and Retrieval Systems (ASRS) [9–13]. Routing algorithms, notably dynamic ones like ACO and GA, play a vital role in optimizing paths within warehouse networks. [14]. ACO, inspired by ant foraging behavior, calculates optimized paths [16]. While GA simulates natural selection to discover optimal solutions for robotic vehicles, determining parameters like speed and movement angle [1]. In this paper, cranes, conveyors, and an Electrified Monorail System (EMS) are chosen as robotic vehicles for algorithm application which play vital roles in lifting, transporting, and conveying goods [7]. The focus of the research is on optimizing algorithms, specifically ACO and GA, for cranes and conveyors in automated warehouses in Malaysia, addressing the challenge of routing optimization in complex environments [8]. The study identifies key challenges in routing optimization within automated warehouses, focusing on efficient routing amidst task scheduling complexities and intricate path planning. Addressing these challenges necessitates sophisticated algorithms capable of dynamically adapting to changing environments and optimizing routes in real-time. The research aims to explore and implement optimization algorithms, specifically ACO and GA, for mobile robot planning in warehouses. Objectives include investigating the features of ACO and GA, implementing them using relevant datasets, and evaluating their performance to contribute to advancements in automated warehouse logistics.

2 Related Works

The Genetic Algorithm, introduced in 1975, mimics natural genetic processes, utilizing a population-based approach to evolve potential solutions through generations. This method has proven successful in warehouse operation optimizations, demonstrating adaptability to non-continuous goal functions and the ability to handle multiple variables [20–22]. Ant Colony Optimization (ACO), inspired by the foraging behavior of ants, is a metaheuristic algorithm widely used in combinatorial optimization problems. Introduced by Marco Dorigo in the early 1990s, ACO employs virtual ants to incrementally construct solutions based on local information and pheromone trails. The algorithm converges towards optimal or near-optimal solutions through an iterative process, effectively balancing exploration, and exploitation in the solution space [23–26]. The literature further delves into mobile crane operation planning, addressing factors such as crane selection based on site conditions, utilizing fuzzy logic, stochastic artificial neural networks, and building information modeling. While specific discussions on ACO and GA in mobile crane operation planning are limited, these optimization techniques have proven valuable in minimizing operational costs, enhancing performance, and ensuring safety in mobile crane applications [27, 28]. The summarized literature review provides a foundation for the research's focus on optimizing routing paths for robotic vehicles in the context of warehouse automation.

2.1 Comparison Study of Previous Researcher Work on Genetic Algorithms

In this section, a comparison study of previous researchers' work is presented, focusing on Genetic Algorithm (GA) techniques. The Table 1 below provides an overview of the selected research papers, their evaluation, methods employed, programs used, and conclusions drawn (Table 2).

Table 1. Comparison on Genetic Algorithm

Author Researcher	Technique Evaluation	Method	Program
GA-based Optimization Method for Mobile Crane Repositioning Route Planning	Propose RPOS for mobile crane repositioning using genetic algorithm	Uses DA and GA for optimizing mobile crane relocation	MATLAB
Path Planning for Multiple AGV Systems Using Genetic Algorithm in Warehouse	Introduces modified genetic algorithm with a single chromosome for enhance performance	Enhanced genetic algorithm with a single chromosome and modified crossover for improved performance	It focuses on the algorithms aspects and simulation results

2.2 Comparison Study of Previous Researcher Work on Ant-Colony Optimization

Table 2. Comparison on Ant-Colony Optimization

Author Researcher	Technique Evaluation	Method	Program
Ant Colony Optimization for Real-World Vehicle Routing Problem	Explores VRP variations, delving into ACO basics & its real-world applications in scenarios like time windows for a grocery chain, pickup and delivery for a distribution company, and an online VRP	Metaheuristic ant colony optimization	-
Automated Lift Planning Method for Mobile Cranes	ACO in lift planning: Crane selection, localization, and path planning	Integration of ACO into automated lift planning processes	Use simulation tools for ACO implementations

3 Methodology

3.1 Research Framework

Figure 1 shows the flowchart of research framework represent the structured approach that that guides the optimization process for robotic vehicle routing in automated warehouses.

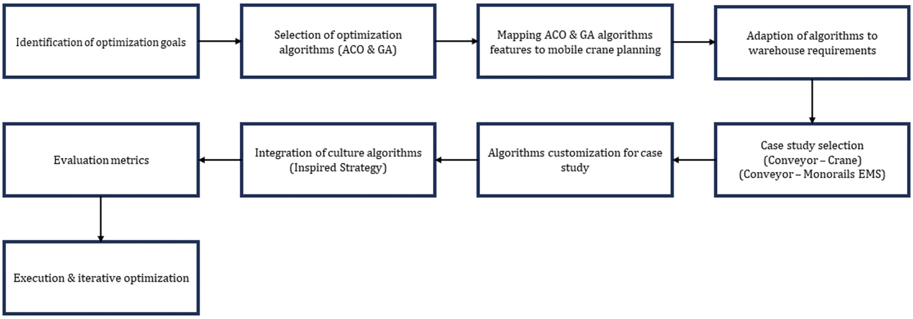

Fig. 1. Research Framework

3.2 Research Activities

A comprehensive overview of the research activities involved in applying the genetic algorithm and ant-colony algorithm to the case studies of robotic vehicle routing optimization in automated warehouses. The steps and strategies employed in each algorithm are outlined below in Fig. 2. The genetic algorithm begins by initializing parameters and creating an initial population of routes. Each route's fitness is evaluated based on optimization objectives like minimizing distance. Selection then picks individuals for the next generation using roulette selection, favoring those with higher fitness. Crossover combines genetic information from two routes to create offspring, while mutation introduces random changes to maintain diversity. The new population is delivered to the belief space, integrating past knowledge. The algorithm terminates when a predefined condition is met, outputting the best route found. Overall, the genetic algorithm efficiently optimizes robotic vehicle routing by evolving routes over successive generations. The Ant-Colony algorithm begins with initialization, where ants are placed at starting locations and initial pheromone levels on route segments are set based on distance factors. Key parameters such as num_ants and num_iterations are defined to influence exploration and convergence. Ants then select the next node using a roulette method based on transfer probabilities derived from pheromone levels and heuristic information. Regular updates to pheromone levels along chosen routes adaptively improve routing choices. An iteration control mechanism ensures convergence, with elite ants guiding others towards optimal solutions. Finally, the algorithm outputs the optimal route based on accumulated pheromone levels, incorporating a fallback approach to handle challenging scenarios.

Fig. 2. Steps of ACO and GA

3.3 Algorithm Implementation

Ant Colony Optimization (ACO) and Genetic Algorithm (GA) implemented using the Python programming language in Spyder software to tackle the challenges associated with optimizing robotic vehicle routing in automated warehouses. The ACO algorithm will undergo modifications to integrate constraints and requirements, focusing on enhancing pheromone update rules and developing effective heuristics for route selection. Leveraging Python libraries such as NumPy and SciPy, the implementation will ensure efficient execution and visualization capabilities through Matplotlib. Similarly, the GA algorithm will be customized to handle the complexities of robotic vehicle routing, involving the design of appropriate chromosome representations, selection of genetic operators (crossover, mutation), and exploration of encoding schemes. Utilizing Python libraries like DEAP or PyGMO, the implementation aims to deliver robust solutions for optimizing robotic vehicle routing and contribute to advancements in warehouse automation.

3.4 Dataset

In this study, data collected from automated warehouses to ensure a comprehensive analysis. The datasets collected from the real-world automated warehouse facility played an important role in the study's subsequent phases. The Warehouse Layout Data provided insights into the physical arrangement of conveyors and crane systems, offering details on positions, CRANE numbers, rack numbers, and position. Task Execution Data, logged the movements of items within the warehouse, including parameters like Source and Destination Nodes, Start and End Positions, and timestamps for each movement. Two case studies further enriched the dataset, focusing on the movement of products from conveyors to cranes and conveyors to monorails (EMS), capturing details such as starting position, ending position, and total time taken. These datasets, encompassing spatial layout and operational dynamics, served as foundational resources for the comparative analysis of routing algorithms.

3.5 Evaluation Metrics

Distance Efficiency

Distance efficiency measure the total distance traveled by the robotics vehicles in the optimized routes, with the objective of minimizing the overall route length. The primary focus is on optimizing the spatial aspects of routing, aiming to reduce the total distance covered. The evaluation metric is defined as the sum of distance traveled by the vehicles to traverse the optimized routes. Let D_{best} represent the total distance for the best route identified by the algorithm. The formula for distance efficiency is shown in (1):

$$D_{best} = \Sigma_{\text{i}=1}^{N-1} distance(node_i, node_{i+1}) \quad (1)$$

where $D_{best} = \Sigma_{\text{i}=1}^{N-1} distance(node_i, node_{i+1})$ Where N signifies the total number of nodes in the route. Distance $(node_i, node_{i+1})$ represents the spatial distance between $node_i$ and $node_{i+1}$. Minimizing indicates more efficient routing and improved distance efficiency in navigating automated warehouse tasks.

Time Efficiency

Time efficiency, as described previously, assesses the time taken by the vehicles to complete their assigned tasks, aiming to minimize task completion time. The combination of these two metrics, distance efficiency and time efficiency, provides a comprehensive analysis of the ACO and GA algorithms' effectiveness in optimizing both spatial and temporal aspects of robotic vehicle routing. The comparison between the algorithms will consider their abilities to minimize D_{best} and the total time taken, contributing valuable information for algorithm selection in warehouse automation scenarios.

4 Results and Discussion

Using ACO and GA on the dataset, independent experiments were conducted for each case study to analyze and compare the outcomes of optimizing robotic vehicle routing within automated warehouses. The data were averaged over several experiment cycles. Conveyor-to-crane and conveyor-to-monorail scenarios were the focus of the tests, which offered detailed insights into algorithmic performance in certain operating environments. Significant variations in route pathways were examined using descriptive statistics, taking nodes and edges into account. The differences between ACO and GA's node prioritization and edge connections were displayed graphically on a graph. The X-Y graph, which showed the node index and processing time, clarified each algorithm's efficiency. Notable differences in Node Index and Edge between ACO and GA were investigated, providing information on how algorithms select routes and priorities positions. Multiple experiments were conducted for each case study, focusing on conveyor-to-crane and conveyor-to-monorail scenarios. The outcomes were analyzed and compared, considering route pathways, node prioritization, and edge connections. Descriptive statistics and graphical representations, including X-Y graphs, were employed to highlight notable differences in algorithmic efficiency, providing insights into the advantages and flexibility of ACO and GA in robotic vehicle routing optimization.

Table 3. Results for ACO and GA

Algorithm	Case Study One	Case Study Two
ACO	Best Distance: 1032 (u) Execution Time: 1.67 (s)	Best Distance: 1071 (u) Execution Time: 1.91 (s)
GA	Best Distance: 1062 (u) Execution Time: 0.08 (s)	Best Distance: 1082 (u) Execution Time: 0.08 (s)

In Table 3, the ACO algorithm demonstrated superior performance in both Case Study 1 and Case Study 2, achieving best distances of 1032 and 1071 units, respectively, compared to GA's distances of 1062 and 1082 units. This signifies ACO's effectiveness in optimizing travel distances within automated warehouses. However, GA highlighted remarkable computational efficiency, completing optimization in 0.08 s for both case studies, while ACO required 1.67 s for Case Study 1 and 1.91 s for Case Study 2. Despite ACO's longer execution times, its ability to yield shorter distances suggests a trade-off between optimization and computational efficiency.

Distance Comparison

In evaluating the performance of the ACO and GA algorithms, the following distance comparisons were observed. The analysis is based on the average results obtained through multiple runs using dataset variations in Python code, ensuring a comprehensive evaluation. Based on the results collected from the implementation, a comparison between ACO and GA is illustrated in Table 4 for both the case studies in terms of distance.

Table 4. Average Distance Comparison for ACO and GA

Algorithm	Case Study One	Case Study Two
ACO	Mean: 1057 Mode: 1032 Standard Deviation: 16.43	Mean: 1074 Mode: 1071 Standard Deviation: 47.20
GA	Mean: 1073 Mode: 1062 Standard Deviation: 11.84	Mean: 1065 Mode: (1062,1072,1090) Standard Deviation: 36.05

Execution Time

In analyzing the performance of both algorithms across multiple instances in each case study, the average execution time is calculated based on multiple runs of the experiments. The mean represents the central tendency, indicating the typical time taken, while the standard deviation provides a measure of the variability or dispersion of the execution times. The reported average execution times and standard deviations capture the trends and variations observed across these multiple runs. The average execution times for both algorithms in each case study are as follows in Table 5:

Table 5. Average Execution Time Comparison for ACO and GA

Algorithm	Case Study One (s)	Case Study Two (s)
ACO	Mean: 2.08 Mode: (2.08, 2.19) Standard Deviation: 0.39	Mean: 1.75 Mode: (1.63, 1.70, 1.73) Standard Deviation: 0.06
GA	Mean: 0.11 Mode: 0.11 Standard Deviation: 0.05	Mean: 0.52 Mode: (0.16, 0.26, 0.62, 0.80) Standard Deviation: 0.05

5 Conclusion

In conclusion, the comparison between Ant Colony Optimization (ACO) and Genetic Algorithm (GA) algorithms in terms of distance and execution time favored ACO, demonstrating its superiority in minimizing travel distances for robotic vehicles in automated warehouses. ACO consistently outperformed GA across both case studies. The study concludes that ACO is more suitable for optimizing automated warehouses due to its effectiveness in enhancing efficiency and productivity. ACO's adaptability to dynamic environments and superior performance compared to GA further solidify its suitability for routing optimization. Factors beyond performance metrics, such as adaptability to dynamic environments, scalability, robustness against uncertainties, ease of implementation and maintenance, and compatibility with existing infrastructure, are crucial when selecting the optimal algorithm for routing optimization in automated warehouses. Recommendations for further experimentation include exploring hybrid approaches, evaluating algorithms under varying conditions, and conducting real-world pilot tests or simulations. Ultimately, the findings of the paper are expected to contribute practically by informing decision-making processes for warehouse operators and logistics managers, enabling them to choose the most appropriate algorithm and improve overall warehouse efficiency.

References

1. Vrcan, Ž, Lovrin, N.: Genetic algorithm based optimisation of conveyor belt material cross section area. Teh. Vjesn. **17**(2), 137–143 (2010)
2. Michael Christofia, E.T., Pereira, V., Tarba, S., Makrides, A.: Artificial intelligence, robotics, advanced technologies and human resource management: a systematic review. Int. J. Hum. Resour. Manag. **33**(6), 1237–1266 (2022)

3. Goel, R., Gupta, P.: A Roadmap to Industry 4.0: Smart Production, Sharp Business and Sustainable Development (2020)
4. Jorge Ribeiro, S.P., Lima, R., Eckhardt, T.: Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review. Procedia Comput. Sci. **181**, 51–58 (2021)
5. van Geest, M., Tekinerdogan, B., Catal, C.: Smart warehouses: rationale, challenges and solution directions. Appl. Sci. **12**(1), 1–16 (2022)
6. Zeinab, S.N.G., Ahmed, E., Saeed, R.A., Mukherjee, A.: Energy optimization in low-power wide area networks by using heuristic techniques. LPWAN Technol. IoT M2M Appl. Acad. Press, no. ISBN 9780128188804, pp. 199–223 (2020)
7. Gwak, H.S., Lee, H.C., Choi, B.Y., Mi, Y.: Ga-based optimization method for mobile crane repositioning route planning. Appl. Sci. **11**(13) (2021)
8. Bell, J.E., McMullen, P.R.: Ant colony optimization techniques for the vehicle routing problem. Adv. Eng. Informatics **18**(1), 41–48 (2004)
9. Sadeghi, M., Nikfar, M., Momeni, F.: Optimizing warehouse operations for environmental sustainability: A simulation study for reducing carbon emissions and maximizing space utilization **4**, 35–44 (2024)
10. Javaid, M., Haleem, A., Singh, R.P., Suman, R.: Substantial capabilities of robotics in enhancing industry 4.0 implementation. Cogn. Robot. **1**(June), 58–75 (2021)
11. Matters, L.: HSS Magazine - Swisslog wins major order from IKEA Supply Malaysia (2019)
12. Pengerang Integrated Complex (PIC) Achieves New Milestones: PETRONAS Media Centre. Retrieved from (2019)
13. Reserve, T.M.: Coca-Cola positive on Malaysia's outlook (2022)
14. Talaviya, T., Shah, D., Patel, N., Yagnik, H., Shah, M.: Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides. Artif. Intell. Agric. **4**, 58–73 (2020)
15. Wahdan, M.: Motion planning for autonomous mobile robots. AEJ - Alexandria Eng. J. **44**(1), 51–57 (2005)
16. Zhang, S., Pu, J., Si, Y., Sun, L.: Path planning for mobile robot using an enhanced ant colony optimization and path geometric optimization. Int. J. Adv. Robot. Syst. **18**(3), 1–15 (2021)
17. Dere, M.E.: Optimum path planning for mobile robots, MS Dissertation. Konya Tech. Univ. (2019)
18. Shi, K., et al.: Path planning optimization of intelligent vehicle based on improved genetic and ant colony hybrid algorithm. Front. Bioeng. Biotechnol. **10**(July), 1–17 (2022)
19. Zhen, L., Li, H.: A literature review of smart warehouse operations management. Front. Eng. Manag. **9**(1), 31–55 (2022)
20. Zhang, H.Y., Lin, W.M., Chen, A.X.: Path planning for the mobile robot: A review. Symmetry (Basel) **10**(10) (2018)
21. Krishnan, E.R.K., Wahab, S.N.: A qualitative case study on the adoption of smart warehouse approaches in Malaysia. E3S Web Conf., vol. 136 (2019)
22. Hassanat, A., et al.: Choosing mutation and crossover ratios for genetic algorithms-a review with a new dynamic approach. Inf. **10**(12) (2019)
23. Yan, Y., Li, Q., Zhang, C., Wang, L., Liao, J.: Recent advances in ant colony optimization and its application in engineering optimization problems. Swarm and Evolutionary Computation **100637**, 54 (2020)
24. Wang, B.L.R., Gao, Y.: A novel ant colony optimization algorithm based on reciprocal strategy and pheromone perturbation. J. Intell. Manuf. **29**, 1521–1534 (2018)
25. S, S., Roy, S.: A hybrid ant colony optimization algorithm for multi-objective economic dispatch problem. Swarm Evol. Comput. **48**, 43–55 (2019)
26. Rodrigues, V.M.C.F.A., Lopes, L.R.: Ant colony optimization for the minimum spanning tree problem with additional constraints. Swarm Evol. Comput. **60**, 100736 (2020)

27. Di Wu, Y.L., Wang, X.: Algorithm of crane selection for heavy lifts. J. Comput. Civ. Eng. **25**(1), 57–65 (2011)
28. Sawhney, A., Mund, A.: IntelliCranes: an integrated crane type and model selection system **19**(2), 227–237 (2001). 2010
29. Guo, H., Zhou, Y., Pan, Z., Lin, X.: Automated lift planning methods for mobile cranes. Autom. Constr. **132**(July), 103982 (2021)